

IN THE SPECIFICATION

Please amend the paragraphs of the specification as follows:

On page 8, please replace the paragraph starting on line 3 with the following paragraph:

FIG. 4 illustrates, in accordance with an embodiment, a communication protocol stack 400 for over the air interface between access terminal 104 and access network 101. The operations of forward channel 200 and reverse channel 300 may be according to communication protocol stack 400. Communication protocol stack 400 may include a physical layer 401, a MAC channel layer 402, a security layer 403, a connection layer 404, a session layer 405, a stream layer 406, and a application layers layer 407. Physical layer 401 provides the channel structure, frequency, power output, modulation, and encoding requirements for forward channel 200 and reverse channel 300. MAC channel layer 402 defines the procedures used to receive and transmit over physical layer 401. Security layer 403 provides authentication and encryption services. Connection layer 404 provides over the air link data connection establishment and maintenance services. Session layer 405 provides protocol negotiation, configuration, and session state maintenance functionality. Stream layer 406 provides multiplexing of distinct applications. Application layers 407 provide default signaling and default packet application for transporting signaling and user data between an access network and an access terminal.

On page 14, please replace the paragraph starting on line 3 with the following paragraph:

FIG. 7 illustrates, in accordance with an embodiment, various states of the Idle State Protocol, which are executed in idle state 701 associated with access network 101, and idle state 752 associated with access terminal 104. The states of the protocol in idle state 752 at access network 101 may include inactive state 851, monitor state 852, sleep state 853, and connection setup state 854. The states of the protocol in idle state 701 at access terminal 104 may include inactive state 801, monitor state 803, sleep state 802, and connection setup state 804. To conserve power at access terminal 104, access terminal 104 and access network 101 maintain sleep states 802, 853. Access network 101 does not send a message to access terminal 104 during the sleep periods, and access terminal 104 does not expect to receive any messages during

the sleep period either, in accordance with an embodiment. Access network 101 may initiate the connection setup by sending a Page message, in accordance with an embodiment, and Access terminal 104 responds with a Connection-Request message. Alternatively, the access terminal 104 may initiate the connection setup by sending a Connection-Request message. The connection setup occurs in the connection setup states 854 and 804 of, respectively, access network 101 and the access terminal 104. If the connection is not denied, access terminal 104 and access network 101 exchange further messages to set-up a connection. The messages may include a Traffic-Channel-Assignment message, ACK message, and Traffic-Channel-Complete message. A successful establishment of a connection results in access terminal 104 being in connected state 753 (shown in Fig. 6), and access network 101 being in connected state 702 (shown in Fig. [[6.]] 6).

On page 16, please replace the paragraph starting on line 14 with the following paragraph:

Access terminal 104 and access network 101 may use communication resources allocated during the setup phase to send and receive data during open states 952, 902. A connection may be in a busy open state or in an idle open state during open states 902, 952, in accordance with an embodiment. When a connection is in a busy open state, data exchange occurs between access network 101 and access terminal 104, either on the forward link, or on the reverse link, or on both. When there is no data to be exchanged, the connection transitions to the idle open state. When data becomes available for the transmission from either access network 101 or access terminal 104, the state of the connection transition from the idle open state to busy open state.

On page 18, please replace the paragraph starting on line 21 with the following paragraph:

FIG. 10 illustrates, in accordance with an embodiment, a flow chart 1100 for use by a resource manager at the access network 101. At step 1101, the resource manager may be in the normal operating state. Normally, several open connections may exist at the same time. Few of the open connections may be in the busy open state, while the others may be in the idle open state. The open connections, in accordance with an embodiment, in the idle state are running

their respective inactivity timers. When a request for opening a new connection arrives, the resource manager checks at step 1102 if any resources are available for allocation. If there are no resources available, the resource manager at step 1103 denies the connection request, and the control flow loops back to step 1101. On the other hand, in accordance with an embodiment, if there are resources available, the resource manager at step 1104 accepts the request for opening a connection, and allocates resources to the new connection in a connection setup routine. Subsequently, the control flow loops back for the resource manager ~~loops-back~~ to step 1101.

On page 23, please replace the paragraph starting on line 12 with the following paragraph:

Generally stated, in accordance with an embodiment, in a communication system for communication of data, a method and apparatus provides for an efficient allocation of communication resources under overload condition. FIG. 12 depicts a flow chart, in accordance with an embodiment, for allocating communication resources to a user, when there are no free resources available. At step 1301, a request is detected for opening a connection for a user for communication of data. At step 1302, an open connection is selected. At step 1303, the selected open connection is released. At step 1304, the communication resources corresponding to resources released based on releasing the selected open connection are allocated to the user. The selected open connection may be in the idle open state or in the busy open state, in accordance with an embodiment. Therefore, the selected open connection<sub>1</sub> in accordance with an embodiment<sub>1</sub> is in an idle open state, and the selected open connection<sub>2</sub> in accordance with another embodiment<sub>2</sub> is in a busy open state.

On page 25, please replace the paragraph starting on line 16 with the following paragraph:

It may be necessary at 1302 to determine whether an open connection is in a busy open state and no open connection is in an idle open state. The selected open connection<sub>1</sub> in accordance with an embodiment<sub>1</sub> is the open connection in the busy open state.

On page 26, please replace the paragraph starting on line 22 with the following paragraph:

It may be necessary at 1302 to determine whether at least an open connection is in the busy open state and at least an open connection is in the idle open state[[ ]]. The selected open connection in accordance with an embodiment may be one of the determined open connections. If the list of the open connections includes two or more open connections in the busy open state and two or more open connections in the idle open state, an open connection is determined from the two or more open connections with the longest idle open state connection time. The selected open connection is the determined open connection with the longest idle open state connection time.

On page 29, please replace the paragraph starting on line 3 with the following paragraph:

FIG. 13 depicts a general block diagram of a controller 1400, in accordance with an embodiment, for controlling connections in access network 101. Controller 1400 may include a connection manager 1401 and a channel resource manager 1402. Connection manager 1401 controls allocation/de-allocation of a number of independent connection controllers 1403A-N. Connection controller 1403 controls various aspects of a connection between access terminal 104 and access network 101. The controlling aspects may include controlling flow of data packets between access terminals 1407A-N and data network 1404. Other controlling aspects may include mobility management, soft handoff, hard handoff, and radio link protocol. Channel resource manager 1402 controls a number of channel resources 1405A-N. Channel ~~resource~~ resources 1405A-N may include data queuing, modulating, demodulating, and decoding functions. In the forward direction, the channel resources 1405A-N may interface with a scheduler 1406. Scheduler 1406 determines which connection to serve and schedules a data unit from ~~resource 1405~~ resources 1405A-N to be transmitted on a time division basis to an access terminal in access terminals 1406A-N. An open connection may be viewed as a connection between access terminal 104 and data network 1404 where a connection controller from connection controllers 1403A-N and a channel resource from resources 1405A-N are assigned to the connection. Channel resource manager 1402 controls allocation/de-allocation (as indicated by dotted lines) of each channel resource in resources 1405A-N, and connection manager

controls allocation/de-allocation (as indicated by dotted lines) of each connection controller in connection controllers 1403A-N, in accordance with an embodiment. When a request for a connection is received, connection manager 1401 assigns a connection controller 1403 to the connection. At this point, the assigned connection controller takes over the controlling aspect of the connection. Connection controller 1403 communicates with channel resource manager 1402 for assigning a channel resource to the connection. Once a resource is assigned, the connection controller communicates directly with the selected resource to set up a connection path from access ~~terminal 1405A-N~~ terminals 1407A-N to data network 1404. The functions performed by each channel resource 1405A-N may include modulating the data for transmission to access terminal on a forward radio link and demodulating/decoding data received on a reverse link. Note that the physical location of the connection manager 1401 and the channel resource manager 1402 may vary depending on the implementation.

On page 31, please replace the paragraph starting on line 1 with the following paragraph:

When all the channel resources are used by open connections and a request for a connection is detected, the channel resource manager 1402 may select one of the connections and the associated assigned resources for release, and assign the released resources to perform functions associated with data flow of the new connection. The selected open connection may be in an idle open state or in a busy open state, in accordance with an embodiment. If more connections are in the open state, a connection may be selected based on the criterion described herein. When a connection is released, the channel resources ~~[[1405]]~~ 1405A-N and the connection controller resources ~~[[1403]]~~ 1403A-N allocated to the connection are released.

On page 31, please replace the paragraph starting on line 1 with the following paragraph:

To determine when an overload condition has been reached, i.e., when there are no more channel resources available for allocation, or when the available channel resources are limited, the channel resource manager 1402 may employ several techniques. A method may include, in accordance with an embodiment, establishing a pre-configured number of maximum connections per channel, which are configured during system installation. When such a number of pre-configured connections has been reached, the channel resource manager 1402 may assume the

channel to be overloaded or has reached the limit. An alternative method or in addition, in accordance with an embodiment, may be to monitor the reverse link loading. When the loading exceeds a certain threshold, the channel may be considered overloaded. In an embodiment, this can be accomplished by monitoring the reverse link busy data bit. When the fraction of time of the busy data bit is set over a predetermined window of time and exceeds a threshold, the channel may be considered overloaded. The threshold may be predetermined. The overload condition or the condition of the limited availability of channel resources may be determined based on other factors. For example, the activity level on the overhead channels such as reverse link pilot channel, or supplemental channels, the data rate control channel, or reverse link power control sub-channel may determine the overload condition. Additionally, or alternatively, the overload condition may be determined based over utilization of the power control channels, or lack of headroom on the power level of the forward link signal.

On page 32, please replace the paragraph starting on line 18 with the following paragraph:

To determine the connection for release according to the described algorithms herein, the channel resource manager 1402 may estimate performance measures such as the connection time (the time duration a connection has been open), amount of data bytes transferred in the forward direction, amount of data bytes transferred in the reverse direction, and the idle time (when there is no data in the forward or reverse direction). These can be collected at the channel resources 1405A-N, and periodically updated to the channel resource manager 1402. Additionally or alternatively, the connection for release may be selected based on a grade of service assigned to a user. The candidates with a low the grade of service may be selected for release in favor of candidates with a high grade of service.

On page 35, please replace the paragraph starting on line 1 with the following paragraph:

The various illustrative logical blocks, modules, and circuits described, in connection with the embodiments disclosed herein, may be implemented or performed with a general purpose processor, a digital signal processor (DSP), an application specific integrated circuit (ASIC), a field programmable gate array (FPGA) or other programmable logic device, discrete

gate or transistor logic, discrete hardware components, or any combination thereof designed to perform the functions described herein. A general purpose processor may be a microprocessor, but in the alternative, the processor may be any conventional processor, controller, microcontroller, or state machine. A processor may also be implemented as a combination of computing devices, e.g., a combination of a DSP and a microprocessor, a plurality of microprocessors, one or more microprocessors in conjunction with a DSP core, or any other such configuration.